GEOSCIENCE AUSTRALIA

Australian Government
Geoscience Australia

AUSTRALIA-INDONESIA FACILITY FOR DISASTER REDUCTION

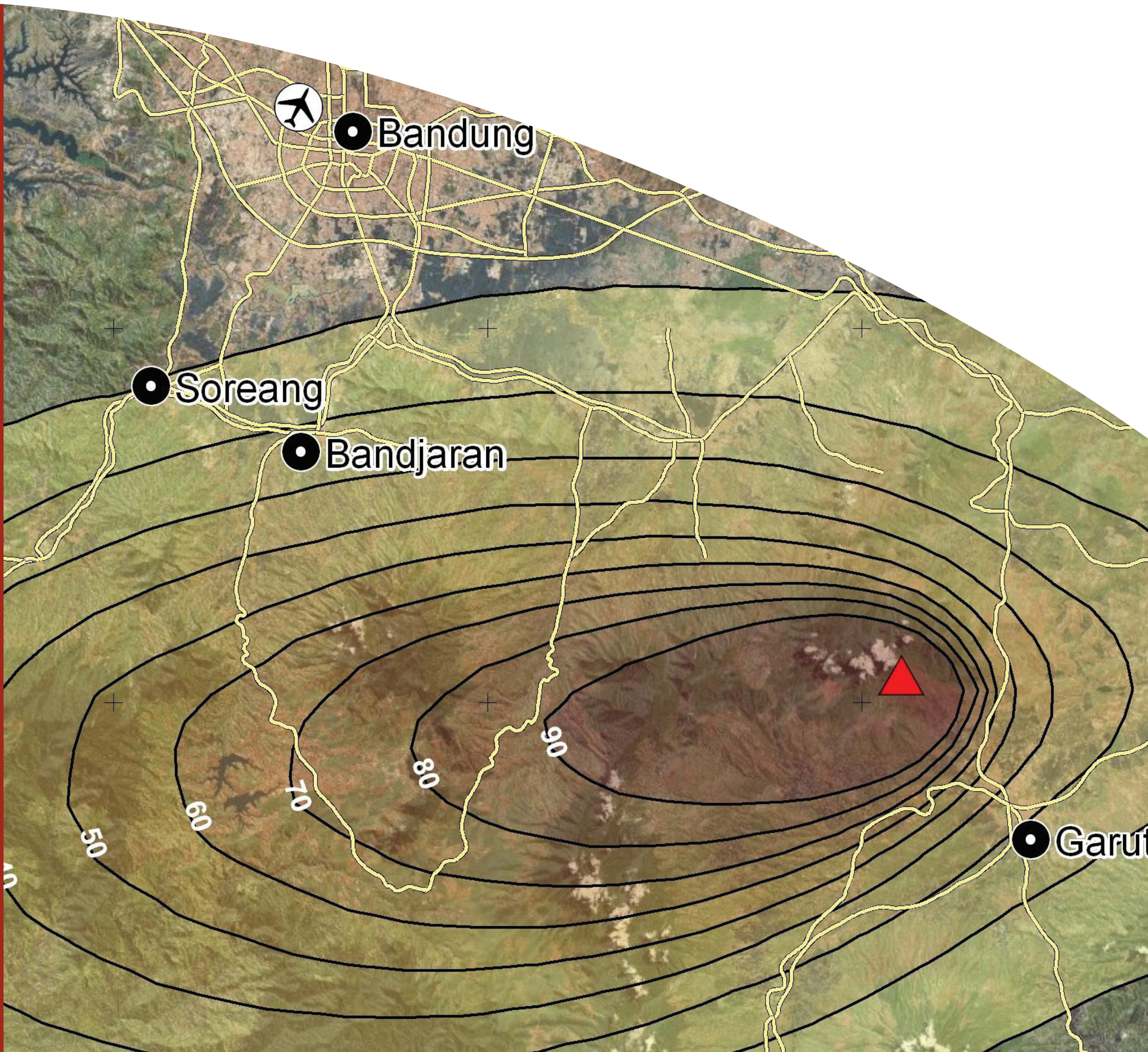
PHIVOLCS DOST

Python-FALL3D: User Manual

A procedure for modelling volcanic ash hazards

Adele Bear-Crozier

Record
2011/33
GeoCat # 71843



APPLYING GEOSCIENCE TO AUSTRALIA'S MOST IMPORTANT CHALLENGES

Python-FALL3D: User Manual

A procedure for modelling volcanic ash hazards

GEOSCIENCE AUSTRALIA
RECORD 2011/33

By

Adele Bear-Crozier[1]

Australian Government
Geoscience Australia

1. Geoscience Australia, Geospatial and Earth Monitoring Division, Risk and Impact Analysis Group

Department of Resources, Energy and Tourism
Minister for Resources and Energy: The Hon. Martin Ferguson, AM MP
Secretary: Mr Drew Clarke

Geoscience Australia
Chief Executive Officer: Dr Chris Pigram

ISSN 1448-2177
ISBN 978-1-921-954-36-8 (Hardcopy)
ISBN 978-1-921-954-35-1 (Web)

GeoCat # 71843

Bibliographic reference: Bear-Crozier, A. N. 2011. Python-FALL3D: User Manual – A procedure for modelling volcanic ash hazards. Geoscience Australia Record No.2011/33

.

Table of Contents

1. Introduction

1.1 PURPOSE

The volcanic ash dispersion model FALL3D (Version 6.2) simulates the fallout of volcanic ash during explosive volcanic eruptions. It is used to understand how volcanic ash interacts with the surrounding atmosphere and where it is deposited at ground level. The purpose of this manual is to introduce a user with no programming or computational modelling experience to FALL3D (Version 6.2) using software called python-FALL3D. Python-FALL3D was developed jointly by Geoscience Australia (GA), the Australia-Indonesia Facility for Disaster Reduction (AIFDR), Badan Geologi (BG) and the Philippines Institute of Volcanology and Seismology (PHIVOLCS). Python-FALL3D features a series of python scripts around the core dispersion model FALL3D (Version 6.2) which simplifies the modelling procedure. The manual features step-by-step instructions for installing and running simulations of volcanic ash fallout using python-FALL3D for deterministic (single scenario), probabilistic (multiple wind) and forecasting purposes.

1.2 SCOPE

This manual provides instructions for installing and running python-FALL3D in a Unix/Linux environment. It incorporates step-by-step instructions for creating volcanological, meteorological and topographic input files, running an eruptive scenario and viewing the results. The package includes two example scenarios based on historical volcanic eruptions in Indonesia which will familiarise new users with the modelling procedure and test if the installation procedure has been successful.

1.3 AUDIENCE

This resource is intended for geoscientists and natural hazard modellers who have a volcanological and/or geological background but no or limited computer programming background.

2. Background

The distribution and thickness of volcanic ash deposited during mildly to highly explosive volcanic eruptions has important life safety, livelihood, economic and political implications for densely populated areas that are affected. A number of computational modelling tools have been developed in recent decades for forecasting the transport and deposition of volcanic ash. Geoscience Australia undertook a study to test and assess existing volcanic ash hazard computational models and evaluate each of these models for different purposes (i.e. single scenario, probabilistic, forecasting). Volcanic ash hazard computational models could be loosely classified into two main groups based on their intended application;

1. Advection-diffusion models which describe particle diffusion transport and sedimentation and can simulate volcanic ash fallout at ground level relative to an eruptive source (e.g. HAZMAP, TEPHRA, FALL3D and ASHFALL).

2. Particle-tracking models which can simulate volcanic ash cloud height and extent at specific times (e.g. PUFF, HYSPLIT and VAFTAD).

2.1 FALL3D

An existing advection-diffusion-sedimentation model has been trialled and adapted for use in South East Asia in response to the needs of government agencies and emergency managers in this region. This model is the widely used, open source volcanic ash hazard model FALL3D (Version 6.2). FALL3D was developed jointly between the Instituto Nationale Geofiscia Vulcanologia (INGV; Italy) and Barcelona Supercomputing Centre (BSC; Spain). FALL3D solves the advection-diffusion-sedimentation equation which governs the settling of ash particles through the atmosphere during a volcanic eruption, including aspects of ground level thickness, load and distribution. It is able to model the transport and deposition of volcanic ash at ground level during an explosive volcanic eruption. It has the ability to model the dispersal of volcanic ash in a wind field that experiences changes in wind speed, direction and air temperature with altitude and over time. FALL3D also considers the interaction between topography and the meteorological conditions and the impact this may have on dispersal of ash at ground level.

2.2 PYTHON-FALL3D – A SIMPLIFIED USER INTERFACE

A Python wrapper was developed jointly between Geoscience Australia (GA) and the Australia-Indonesia Facility for Disaster Reduction (AIFDR) which modifies the modelling procedure of FALL3D to simplify its use for those with no background in computational modelling. Three modelling procedures are available through a unified interface: scenario-based modelling (single event), hazard mapping (probabilistic wind) and forecasting (predictive). Python-FALL3D outputs are geospatially referenced in a standard format and can be viewed alongside other datasets important for impact and risk analysis such as: population density, exposure of the built environment and crop extents. The hazard maps produced contour connecting points of equal volcanic ash thicknesses (or ash load: mass per unit area; or ash concentration: volume per unit area). Each map may contain contours of volcanic ash thicknesses/load that vary in appearance according to the volcanological and meteorological conditions during the eruption. Collectively these hazard maps are intended for use by government agencies to assess the risk of volcanic ash for communities.

Validation of the underlying numerical model (FALL3D) against observed data from known historical eruptions in the South East Asian region was an important part of the two-year development stage for python-FALL3D. Validation, a measure of how accurately the model reproduces known volcanic ash deposits has important implications for the expected uncertainty in modelled outputs and the relative sensitivity of different input parameters (i.e. wind speed versus ash grainsize) FALL3D has been validated with a few specific examples from volcanic eruptions in Indonesia and Papua New Guinea.

3. Useful UNIX commands

Python-FALL3D is designed to run in a UNIX/Linux environment such as Ubuntu Linux. Although directories and output files can be viewed and manipulated through the windows manager the user is still required to run the model from a UNIX command line using a terminal window. The user is therefore required to know a number of basic UNIX commands. There are eight commands which are particularly useful when for navigating through a UNIX environment using python-FALL3D:

- **cd** *<directory name>*
  change directory
  (Open this directory)

- **cd .. -**
  go up one directory
  (Close this directory and open the parent directory)

- **cd ../..** –
  go up two directories
  (Keep adding "**/..**" to go up more than two directories)

- **ls -l**
  list
  (Display contents of current directory)

- **pwd**
  print working directory
  (Display current location)

- **cp** *<filename> <directory>*
  copy this file and move it to **this directory**

- **cp *.**<*extension*> *<directory>*
  copy all files with this extension and move them to **this directory**

- **mkdir** *<directory>*
  make directory
  (Make a new directory (folder) at this location – this is followed by a space and the name of the new directory)

Other commands that the user may require to use python-FALL3D include:

- **svn co**
  checkout
  (Refers to 'checking out' a repository, scripts etcetera)

- **python –**
  (This is then followed by a space and the name of the python script that the user would like to run)

- **ln –s**
  link
  (Allows you to create a shortcut to a specified directory from the current directory)

4. System requirements & dependencies

To run python-FALL3D you will require the following:

- A standard PC with at least 4GB of RAM and an Ubuntu Linux operating system (see **http://www.ubuntu.com** for instructions on downloading and setting up Ubuntu Linux – freely available); and
- An internet connection (for initial download and installation only unless specified).

4.1 DOWNLOADING DEPENDENCIES

Seven dependency programs are required for python-FALL3D to run successfully. The user must configure Ubuntu's Synaptic manager so that it will be able to locate and install these programs (internet connection required) prior to installing python-FALL3D.

1. Open Ubuntu Linux and ensure an internet connection is established.

2. Select '**System**' from the toolbar menu and then select '**Administration**' and then '**Synaptic Package Manager**' to open a new window.

3. Select the tab labelled '**Repositories**' and tick all the box options (if not already checked).

4. Close Synaptic Package Manager.

4. Select '**Applications'** from the toolbar menu and the select '**Accessories'** and then '**Terminal'** to open a new terminal (Follow this procedure whenever a new terminal is needed).

5. To download the first dependency program called 'subversion' type:

sudo apt-get install subversion

6. Press Enter

Subversion will be downloaded and installed automatically.

7. Repeat this procedure for the 6 remaining dependency programs listed below:

sudo apt-get install python-numpy

sudo apt-get install python-scientific

sudo apt-get install gfortran

sudo apt-get install python-gdal

sudo apt-get install gdal-bin

sudo apt-get install libnetcdf-dev

5. First time installation of python-FALL3D

5.1 INSTALLING PYTHON-FALL3D

Instructions for installing python-FALL3D onto your PC for use in a linux/UNIX environment are detailed below:

- Green text highlights the UNIX commands that are used;
- Blue text indicates a pathway of directories to be taken and;
- Red text indicates single directories, file names, websites, programs and usernames.

You will only need to follow this step once for initial setup purposes. It details how to create a sandpit where python-FALL3D will be installed and run. The example below provides suggested names for newly created directories highlighted by the symbols “<” and “>”. **Do not type the symbols “<” and “>”.**

1. Open a new terminal (double click on the display icon on the desktop).

2. To create a sandpit type:

mkdir <**sandpit**>

(e.g. **mkdir sandpit**)

3. To change directory into your sandpit type:

cd <**sandpit**>

4. To download python-FALL3D type:

svn co --username anonymous http://www.aifdr.org/svn/aim/branches/fall3d_v6 aim

5. When prompted for a password press ‘Enter’ (no password necessary).

6. To change to the python-FALL3D source code directory type:

cd aim/source/aim

7. To install python-Fall3D type:

python install_fall3d.py

8. When prompted with 'update .bashrc file (Y or N) type:

Y

The installation of python-FALL3D is complete.

The location of the output data is controlled by the environment variable called **TEPHRADATA**. It is specified in the system file named **.bashrc** in your home directory. If you want the output data to be stored elsewhere you can edit the **.bashrc** file using the following procedure:

9. Open a new terminal and navigate to your home directory.

10. Type **gedit .bashrc** (or use your preferred editor).

11. The **.bashrc** file will open.

12. Scroll down to the line:

export TEPHRADATA=/<home>/<username>/<tephra>

13. Customise the pathway for output data to be stored when using python-FALL3D.

The default will be **'/<home>/<username>/<tephra>'** and this will be used for all future reference to the **TEPHRADATA** area throughout this manual.

14. Save and close the terminal window.

Note: It is important to close this terminal window to ensure that the environment variables set by the installation process come into effect.

5.2 TESTING PYTHON-FALL3D

There is a script called **test_all.py** which will test if the installation was successful. To run the script:

1. Open a new terminal.

2. Change to the directory:

cd <sandpit>/aim/testing

3. To run the test script type:

python test_all.py

5.3 VALIDATION SCENARIOS

Python-FALL3D has been validated against a number of historical eruptions in order to ensure the modelled outputs accurately reproduce observed ash thickness and loads. Two validation scenarios are included with the installation of python-FALL3D; the 1840 eruption of Gunung Guntur, Indonesia and the 1994 eruption of Tavurvur Volcano, Papua New Guinea. It is important that users run each validation and compare the generated outputs with stored model outputs included in reference data as part of the python-FALL3D installation. This serves to familiarise the new user with the modelling procedure and verify that the installation of python-FALL3D works as intended.

5.3.1 Validation Scenario 1 – 1840 eruption of Gunung Guntur, Indonesia

This scenario was developed to validate python-FALL3D against observed ash thicknesses from the 1840 eruption of Gunung Guntur. The scenario was developed by Nugraha Kartadinata (BG), Anjar Heriwaseso (BG), Adele Bear-Crozier (GA), Ole Nielsen (AIFDR), Antonio Costa (INGV), Arnau Folch (BSC) and Kristy Van Putten (AIFDR) at a workshop held at the AIFDR in Jakarta in July 2010. Modelled outputs were compared against observed volcanic ash thickness measured in the field at Gunung Guntur by N. Kartadinata and (internal BG publication).

To run the 1840 Gunung Guntur validation scenario:

1. Open a new terminal.

2. Change to the directory:

cd <sandpit>/aim/validation/guntur

3. To run the Guntur 1840 scenario type:

python guntur1840.py

4. To view model results navigate to TEPHRADATA:

cd /<home>/<username>/<tephra>/guntur1840

5. Compare model output with stored model output for the Guntur 1840 eruption located in the directory below and shown in **Figure 1**;

cd <sandpit>/aim/validation/guntur/reference_data/model_ouputs

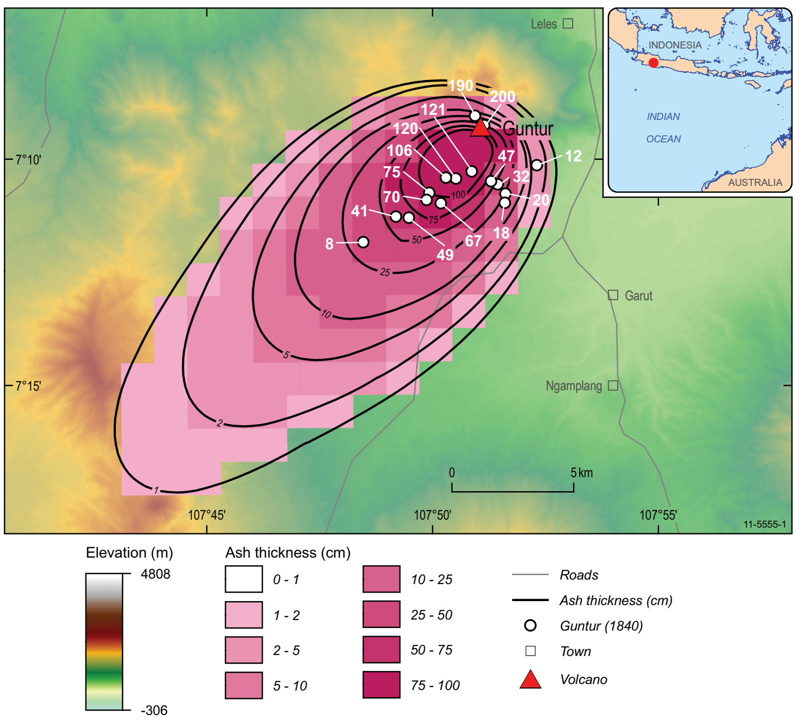


Figure 1 – Stored model output for the 1840 eruption of Gunung Guntur showing good agreement with observed ash thicknesses collected at 16 localities: White points = measured observed ash thicknesses (cm) from G.Guntur (N. Kartadinata); Black lines = ash thickness (cm) isopach map generated by FALL3D; Pink = ash distribution (thickness in cm) generated by FALL3D used to construct isopach map.

5.3.2 Validation Scenario 2 – 1994 eruption Tavurvur Volcano, Papua New Guinea

This scenario was developed to validate FALL3D against observed ash thicknesses from the 1994 eruption of Tavurvur Volcano, East New Britain, Papua New Guinea by James Goodwin (GA) and Adele Bear-Crozier (GA; Goodwin and Bear-Crozier, in prep). Modelled outputs were compared against ash thickness observations collected within the nearby township of Rabaul (destroyed during the eruption) published by Blong and McKee (1995) and Blong (2003).

1. Open a new terminal.

2. Change to the directory:

cd <sandpit>/aim/validation/tavurvur

3. To run the Tavurvur 1994 scenario type:

python tavurvur.py

4. To view model results navigate to TEPHRADATA:

cd /<home>/<username>/<tephra>/tavurvur

5. Compare model output with stored model output for the Guntur 1840 eruption located in the directory below and shown in **Figure 2**;

cd <sandpit>/aim/validation/tavurvur/reference_data/model_ouputs

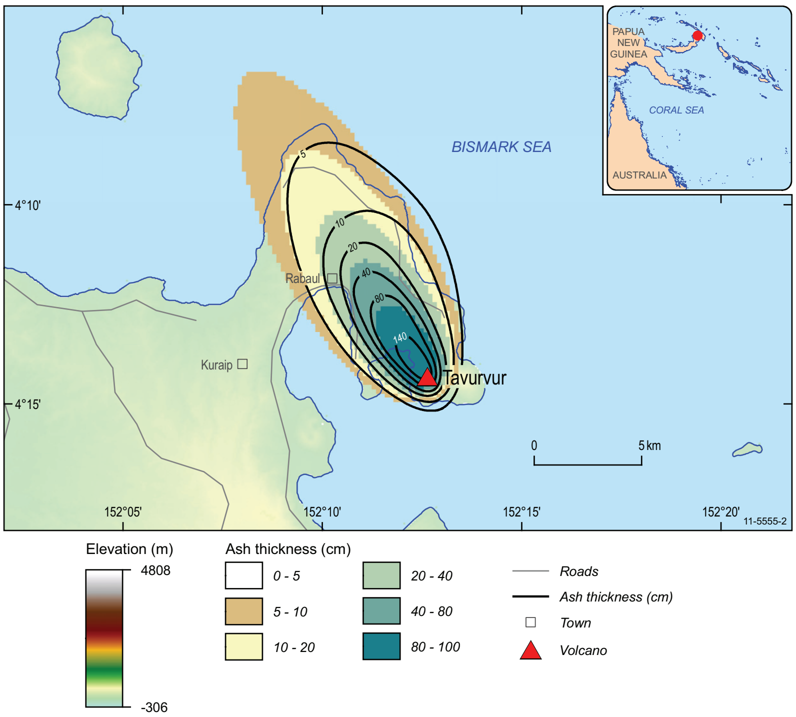


Figure 2 – Stored model output for the 1994 eruption of Tavurvur Volcano showing good agreement with observed ash thickness isopach map produced by Blong and McKee (1995; black lines).

6. Setting up a modelling area

Python-FALL3D has now been successfully downloaded and installed. The validation scripts have been run to test the success or failure of that installation process. Each new user must now set up a volcanic ash modelling area. This is the directory where the user will edit all scripts and run the model. This volcanic ash modelling area will sit within the sandpit but separate from python-FALL3D source code, the test scripts and the validation scenarios (the directory named '**aim**'). You will only need to follow this step once for initial setup purposes.

6.1 BUILDING A VOLCANIC ASH MODELLING AREA

1. Open a new terminal.

2. Change directory into your sandpit:

cd <**sandpit**>

3. To create a modelling area type:

mkdir <**volcanic_ash_modelling**>

A directory named <**volcanic_ash_modelling**> has now been created in the sandpit and is ready to be populated with python scripts from the templates directory.

6.2 TEMPLATE SCRIPTS

The templates directory contains example scripts which the user can copy into their modelling area edit and run as needed. There are three template scripts:

1. **extract_windprofiles.py** – create wind profiles (Table 1)
2. **volcano.py** – run FALL3D (Table 2)
3. **create_hazard maps.py** – create probabilistic hazard map (Table 3)

To copy these scripts to the modelling area:

1. Open a new terminal.

2. Change into the directory:

cd <**sandpit**>**/aim/templates**

3. To view a list of the template scripts type:

ls -l

4. To copy these scripts into a new modelling area type:

cp *.py /<**sandpit**>**/**<**volcanic_ash_modelling_directory**>

All files with the extension '.py' will be copied into the volcanic ash modelling area specified by the user. These files can then be opened, edited and run as needed.

To confirm that the modelling area has been populated with the three template scripts:

1. Open a new terminal.

2. Change into the directory:

cd <sandpit>/<volcanic_ash_modelling>

3. Type:

ls –l

A list of template scripts will appear ready for use.

7. Preparing Input Data

7.1 PREPARING DIGITAL ELEVATION DATA

Python-FALL3D requires a digital elevation model (DEM) and accompanying projection file. Digital elevation models must be in ESRI ASCII format.

- Use the template in Appendix 1 to format a digital elevation model and accompanying projection file compatible with python-FALL3D.

7.2 PREPARING METEOROLOGICAL DATA

Python-FALL3D requires a meteorological input. Two freely available options are currently available to users:

- NCEP1-reanalysis - historical wind conditions (1948 to present) and;
- ACCCES-T - forecasted wind conditions (72 hr).

To download NCEP1-reanalysis data:

- Refer to download instructions in Appendix 2 (internet connection required).

To use ACCESS-T data:

- Refer to Table 2 for web link (python-FALL3D will download automatically; internet connection required).

7.3 INPUT VARIABLES FOR PYTHON SCRIPTS

The three python scripts: '**extract_windprofiles.py**', '**volcano.py**' and '**create_hazard map.py**' are used individually or in combination depending on the modelling procedure chosen. **Tables 1 – 3** provide descriptions of the input variables required for each script.

Table 1 – Description and input options (where applicable) for each input parameter in the python script '**extract_windprofiles.py**'

Input parameter	Description	Units/input options/examples
Location in UTM coordinates of the vent These coordinates will be used to extract a vertical wind profile(s) at a location closest to the vent using NCEP1-reanalysis data from the National Oceanic and Atmospheric Association (Appendix 2).		
vent_easting	location of the vent	UTM coordinates
vent_northing	location of the vent	UTM coordinates
vent_zone	UTM zone of the vent	
vent_hemisphere	hemisphere of the vent	Options: N or S

Time to start extraction The extraction start time indicates when the wind profile will begin and is usually the same time as he start of the eruption.		
start_year	start year of the wind profile	YYYY
start_month	start month of the wind profile	Options: 1, 2 . . . 12
start_day	start day of the wind profile	Options: 1, 2 . . . 31
start_hour	start hour of wind profile	Options: 0, 6, 12 or 18
Time to end extraction The extraction end time indicates when the wind profile will end and must be at least one hour after the 'post_eruptive_settling_duration' to ensure all the simulated volcanic ash has been deposited at ground level (See Table 2)		
end_year	end year of the wind profile	YYYY
end_month	end month of the wind profile	Options: 1, 2 . . . 12
end_day	end day of the wind profile	Options: 1, 2 . . . 31
end_hour	end hour of the wind profile	Options: 0, 6, 12 or 18
Path to directory of NCEP files This will indicate the pathway to the directory where the NCEP input data is stored. This directory should contain the following: **TMP.nc**, **HGT.nc**, **UGRD.nc** and **VGRD.nc**		
NCEP_dir	/<**home**>/<**username**>/<**tephra**>/<**NCEP**> (a directory containing NCEP files)	
Path to directory of generated wind profiles This will indicate the pathway to the directory where the multiple wind profiles generated from the NCEP data will be stored.		
windfield_dir	/<**home**>/<**username**>/<**tephra**>/<**merged**>' (a directory of merged profiles) /<**home**>/<**username**>/<**tephra**>/<**multiple**>' (a directory of multiple profiles)	
Wind field type: option are "merged" or "multiple"		
wind_field_type	This script can produce two types of wind profiles depending on the modelling procedure chosen.	Scenario-based: 'merged' Hazard Map: 'multiple'

Table 2 – Description and input options (where applicable) for each input parameter in the python script '**volcano.py**'

Input parameter	**Description**	**Units/input options/examples**
Short eruption comment to appear in output directory		
eruption_comment	Name of output directory (added to timestamp)	
Temporal parameters (Hours) The wind profile determines the time limit of the simulated eruption (i.e. a 16 hour wind profile means the eruption can't exceed 16 hours). The user must input the eruption start time relative to this wind profile in hours (i.e. 0 = eruption and wind begin together; 1 = eruption begins 1 hour after wind).		

eruption_start	Start time of the eruption: given as the number of hours since time 0 hours.	Default: 0
eruption_duration	Duration of the eruption: given as a number of hours.	
post-eruptive_settling_duration	Duration of post-eruption ash settling: given as a number of hours.	

Location (Volcanological input file)
The topography of the volcano and surrounding area are automatically read into python-FALL3D and the user is only required to define the vent location within that topography in UTM coordinates.

x_coordinate_of_vent	x coordinate (UTM) of the vent location *(UTM zone implied by topography file)*	e.g. 439423
y_coordinate_of_vent	y coordinate (UTM) of the vent location *(UTM zone implied by topography file)*	e.g. 9167213

Vertical discretisation of the model domain
The topography is used to define the horizontal extent of the modelled area in the x and y directions. Vertical discretisation determines the vertical extent of the area being modelling in the z direction. In combination they define the 3-dimensional space into which an eruption column is generated.

z_min	Minimum altitude of vertical domain.	Units: m
z_max	Maximum altitude of vertical domain *(must be greater than the eruption column height).*	Units: m
z_increment	Division of vertical domain into layers for volcanic ash to disperse *(usually 1/10th the z_max; i.e. z_max=10,000 then z_increment = 1000).*	Units: m

Meteorological input
There are three possible wind profile types: (1) NCEP 'merged'; (2) NCEP 'multiple' and; (3) ACCESS-T. The meteorological input will indicate where the wind data is stored as either a single profile (merged), a directory of multiple profiles (multiple) or a website link for the download of online forecasts (ACCESS-T).

wind_profile	'/<**home**>/<**username**>/<**tephra**>/<**merged**>/<**merged.profile**>' (single profile) OR '/<**home**>/<**username**>/<**tephra**>/<**multiple**>' (a directory of multiple profiles) OR 'ftp://ftp-newb.bom.gov.au/register/sample/access/netcdf/ACESS-T/pressure/'

Terrain model
The user must specify which topographic file to use (DEM) by providing the pathway to the file. Python-FALL3D will automatically read in the accompanying projection file (**topography.prj**). In this way the user can utilise a collection of DEM's at varying spatial resolutions.

topography_grid	'/<**home**>/<**username**>/<**tephra**>/<**dems**>/<**topography.txt**>'

Granulometry

The grainsize data should be based on quantitative analysis of volcanic ash samples for the volcano being modelled or a suitable analogue (Bonadonna and Houghton, 2005; Carey and Sigurdsson, 1982). The values below will be derived from sieve data and calculation of the Inman parameters for grainsize distribution and sorting of volcanic ash deposits.

grainsize_distribution	Gaussian (modal) or Bi-Gaussian (bi-modal) grainsize distributions can be modelled.	Options: GAUSSIAN or BIGAUSSIAN
number_of_grainsize_classes	Number of particle classes python-FALL3D will generate.	Default: 10
mean_grainsize	Calculated average grainsize.	Units: phi
sorting	calculated degree of sorting of volcanic ash particles.	
minimum_grainsize	Calculated minimum grainsize.	Units: phi
maximum_grainsize	Calculated maximum grainsize.	Units: phi
density_minimum	Analytically determined density minimum.	Units: kg/m3
density_maximum	Analytically determined density. maximum.	Units: kg/m3
sphericity_minimum	Analytically determined sphericity. minimum *(how rounded are the volcanic ash particles).*	Value between 0 and 1.
sphericity_maximum	Analytically determined sphericity maximum.	Value between 0 and 1.

Source

The source section is where the user defines the eruption style and magnitude (mildly explosive – highly explosive) by specifying the column height and/or the mass eruption rate (Carey and Sparks, 1986; Legros, 2000; Pyle, 1989; Sulpizio, 2005). FALL3D uses one of three source models for eruption generation: the possibilities are ‘point’, ‘suzuki’ or ‘plume’ (Appendix 6). The user is required to input different parameters depending on the source model chosen (see below).

vent_height	Height of the vent above sea level.	Units: m
source_type	Models for eruption generation.	Options: ‘point’, ‘suzuki’ or ‘plume’
mass_eruption_rate	The rate at which magma is ejected from the vent (eruption intensity).	Units: kg/s Options: number or ‘estimate’
height_above_vent	Height of the eruption column.	Units: m Options: number or ‘estimate’
A	Empirically derived suzuki parameter for the position of neutral buoyancy with respect to column height. The greater the value for ‘A’ the higher the mass sits in the simulated column.	**Suzuki Only** Options: values typically between 1 and 4; where; 1 = Strombolian and 4 = Plinian

L	Empirically derived suzuki parameter for the spread of mass within the column with respect to the neutral buoyancy level. The greater the value for 'L' the more horizontally dispersed across the column the mass will be.	**Suzuki Only** Options: values typically between 1 and 5; where; 1 = Plinian and 5 = Strombolian
height_or_MFR	The plume model only requires the user to enter a column height (height) or a mass eruption rate (MFR). It will calculate the other independently.	**Plume only** Options: 'height' or 'MFR'
MFR_minimum	Minimum mass eruption rate.	**Plume only** Units: kg/s
MFR_maximum	Maximum mass eruption rate.	**Plume only** Units: kg/s
exit_velocity	Magma exit speed.	**Plume only** Units:m/s
exit_temperature	Magma exit temperature.	**Plume only** Units:K
exit_volatile_fraction	Volatile fraction (what percentage of the melt is H_2O, CO_2 etc).	**Plume only** Units:%

FALL3D
This section is where the user sets the parameters for volcanic ash dispersal through the atmosphere following the initial eruption. FALL3D uses one of three terminal velocity models for the settling over volcanic ash through the atmosphere: the possibilities are 'ARASTOOPOR', 'GANSER', 'WILSON' and 'DELLINO' (Appendix 6).

terminal_velocity_model	Model for volcanic ash settling through the atmosphere.	Options: 'ARASTOOPOR', 'GANSER', 'WILSON' and 'DELLINO'
vertical_turbulence_model	Vertical turbulence experienced by the ash particles can be user-defined (CONSTANT) or derived from the wind profile (SIMILARITY).	Options: 'CONSTANT' and 'SIMILARITY'
horizontal_turbulence_model	Horizontal turbulence experienced by the mass in the column can be user-defined (CONSTANT) or derived from the wind profile (RAMS).	Options: 'CONSTANT' or 'RAMS' (If vertical turbulence is 'CONSTANT' then 'CONSTANT' else 'RAMS')
vertical_diffusion_coefficient	Mixing of particles vertically within the simulated eruption column.	Only defined by user if vertical and horizontal turbulence is 'CONSTANT' else derived from wind profile (i.e. SIMILARITY/RAMS) Options: High column (1-50) and Low column (50-1000)

horizontal_diffusion_coefficient	Mixing of particles horizontally within the simulated column.	Only defined by user if vertical and horizontal turbulence is 'CONSTANT' else derived from wind profile (i.e. SIMILARITY/RAMS) Options: 1000 – 10000
value_of_CS	A constant value between 0.135 and 0.32 only used when horizontal turbulence is RAMS.	RAMS only

Contouring: True, False, number or list of numbers
Python-FALL3D produces maps of volcanic ash thickness and volcanic ash load. The model contours the ash thickness and load values for viewing in Google Earth (kml) and ArcGIS (shp). There are four options for determining the interval between contours: '**True**': the model will determine equally spaced contours based on the spread of data (a good first approximation); '**False**': no contours; '**Number**': the user can specify the number of contour intervals and the model will generate that number of contours based on the spread of data; and **'List of Numbers**': the user can specific the number of contour intervals and the corresponding value for each interval (user for a standardised classification scheme and comparing different scenarios).

thickness_contours	Type of contouring.	Options: True, False, Number or List of Numbers
load_contours	Type of contouring required for volcanic ash load (kg/m^2).	Options: True, False, Number or List of Numbers
thickness_units	Ash thickness units.	Units: mm/cm/m

Run model using specified parameters (*Procedure 2 - 'Hazard Map' only; Refer to 7.2*)
Location of multiple wind fields for probabilistic hazard mapping and location of generated outputs (one hazard scenario per wind field).

windfield_directory	'/<**home**>/<**username**>/<**tephra**>/<**multiple**>' (a directory of multiple profiles)
hazard_output_directory	'/<**home**>/<**username**>/<**tephra**>/<**hazard_outputs**>' (directory where multiple model runs are to be stored)

Table 3 – Description and input options (where applicable) for each input parameter in the python script '**create_hazard_map.py**'

Input parameter	Description	Units/input options
Vent location in geographic coordinates (decimal degrees)		
vent_easting	Location of the vent.	UTM coordinates
vent_northing	Location of the vent.	UTM coordinates
vent_zone	UTM zone of the vent.	
vent_hemisphere	Hemisphere of the vent.	Options: N or S
Values Hazard maps are based on multiple scenarios and a specified threshold of volcanic ash (kg/m^2). The resulting maps contour the probability of exceeding that specified ash threshold given the multiple scenarios. One map is produced for each threshold value.		
load_values	Volcanic ash load threshold values (kg/m^2) which will be used to generate a hazard map. A separate hazard map will be generated for each load value.	Options: a single threshold value i.e. 0.1, or a list of thresholds values separated by commas and enclosed in square brackets i.e. [0.1, 10, 20]. The resulting map(s) will contour probability of exceeding the ash load threshold in %.
fl_values	Volcanic ash concentration threshold values (kg/m^3) which will be used to generate a hazard map. A separate hazard map will be generated for each concentration value.	Options: a single threshold value i.e. 0.002 or a list of thresholds values separated by commas and enclosed in square brackets i.e. [0.0002, 0.002]. The resulting map(s) will contour probability of exceeding the ash concentration threshold in %.
Contours The model contours probability of exceedance (%) and/or change through time (hour) for viewing in Google Earth and ArcGIS. There are four options for determining the interval between contours: '**True**', '**False**', '**Number**' or '**List of numbers**' (Refer to Table 2).		
ISOCHRON_contours	Contour interval type.	Options: True, False, Number or List of Numbers
ISOCHRON_units	Contour units.	Units: hours
PLOAD_contours	Contour interval type.	Options: True, False, Number or List of Numbers
PLOAD_units	Contour units.	Units: percent
Location of generated windprofiles, hazard map and contours This directory should contain the multiple scenario outputs produced by **volcano.py** which will be used to create the hazard map(s). The hazard map(s) will be stored here.		
model_output_directory	'/<**home**>/<**username**>/<**tephra**>/<**hazard_outputs**>' (a directory of multiple profiles)	

8. Modelling Procedure

There are three modelling procedures available to users of python-FALL3D;

1. Scenario-based,
2. Hazard Map and;
3. Forecasting.

A description of each modelling procedure, the necessary input data, generated outputs and python scripts to be used are detailed in **Table 4**.

8.1 SCENARIO-BASED PROCEDURE

This procedure details how to run a volcanological scenario using a single 'merged' wind field extracted from NCEP1-renalysis meteorological data – a deterministic approach.

1. Open a new terminal.

2. Navigate to your volcanic ash modelling directory

cd <**sandpit**>/<**volcanic_ash_modelling**>

3. Open **extract_windprofiles.py** using a text editor by typing:

gedit extract_windprofiles.py (or use preferred editor)

4. Edit the input variables (**Table 1**).

5. Save and close.

To run type:

python extract_windprofiles.py

6. Open **volcano.py** using a text editor by typing:

gedit volcano.py

7. Edit the input variables (**Table 2**; **Appendix 3**)

8. Rename the script when saving and close.

(e.g **merapi.py**)

9. To run type:

python <volcano>.py

(eg. **python merapi.py)**

Table 4 – Overview of python-FALL3D modelling procedures, input requirements, outputs and python scripts.

Procedure	Description	Input	Output	Python scripts
Scenario-based	A procedure used to model a volcanological scenario and a single 'merged' wind profile. This procedure is useful for deterministic modelling and is not computationally intensive.	1. DEM 2. Merged vertical wind profile (NCEP1) 3. Volcanological scenario	Volcanic ash thickness (mm,cm,m) based on one historical wind profile. Volcanic ash load (kg/m^2) based on one historical wind profile.	**1. extract_windprofiles.py** (Table 1) **2. volcano.py** (Table 2)
Hazard Map	A procedure used to model a volcanological scenario and 'multiple' wind profiles. This procedure is useful for probabilistic assessments based on changing wind conditions but is computationally time consuming.	1. DEM 2. Multiple vertical wind profiles (NCEP1) 3. Volcanological scenario	Probability of exceedance (%) of a volcanic ash threshold value in kg/m^2 based on multiple historical wind profiles.	**extract_windprofiles.py** **volcano.py** **create_hazard_maps.py** (Table 3)
Forecasting	A procedure used to model a volcanological scenario with forecast wind data.	1. DEM 2. Vertical wind profile (ACCESS-T) 3. Volcanological scenario	Volcanic ash thickness (mm,cm,m) based on a forecasted wind profile. Volcanic ash load (kg/m^2) based on a forecasted wind profile.	**volcano.py**

Outputs files are generated for volcanic ash thickness and load each simulated hour in ASCII, grd, shp and kml (Google Earth) format (**Figure 3**).

To view output files navigate to the **TEPHRADATA** area:

cd /<home>/<username>/<tephra>/<volcano>

8.2 HAZARD MAPPING

This procedure details how to run a volcanological scenario using 'multiple' wind fields extracted from NCEP1-renalysis meteorological data - a probabilistic approach. The results of each scenario are merged into a single hazard map showing probability (%) of exceeding a user defined volcanic ash load threshold (kg/m^2). Multiple hazard maps can be generated for multiple ash threshold values.

1. Open a new terminal.

2. Navigate to your volcanic ash modelling directory

cd <**sandpit**>/<**volcanic_ash_modelling**>

3. Open **extract_windprofiles.py** using a text editor by typing:

gedit extract_windprofiles.py (or use preferred editor)

4. Edit the input variables (**Table 1**).

5. Save and close.

To run type:

python extract_windprofiles.py

6. Open **volcano.py** using a text editor by typing:

gedit volcano.py

7. Edit the input variables (**Table 2**; **Appendix 3**).

8. Rename the script when saving and close.

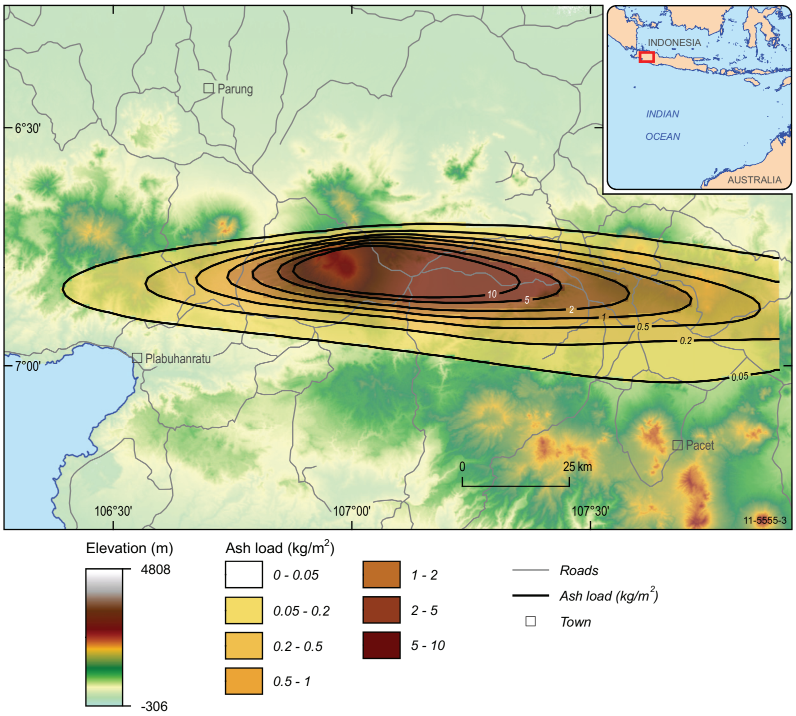
(e.g **merapi.py**)

The script can be run in serial (one computer) or in parallel (multiple nodes; **Appendix 5**).

9. To run the script serially type:

python volcano_multiple_wind.py
(eg. **python merapi_multiple_wind.py**)

A.



B.

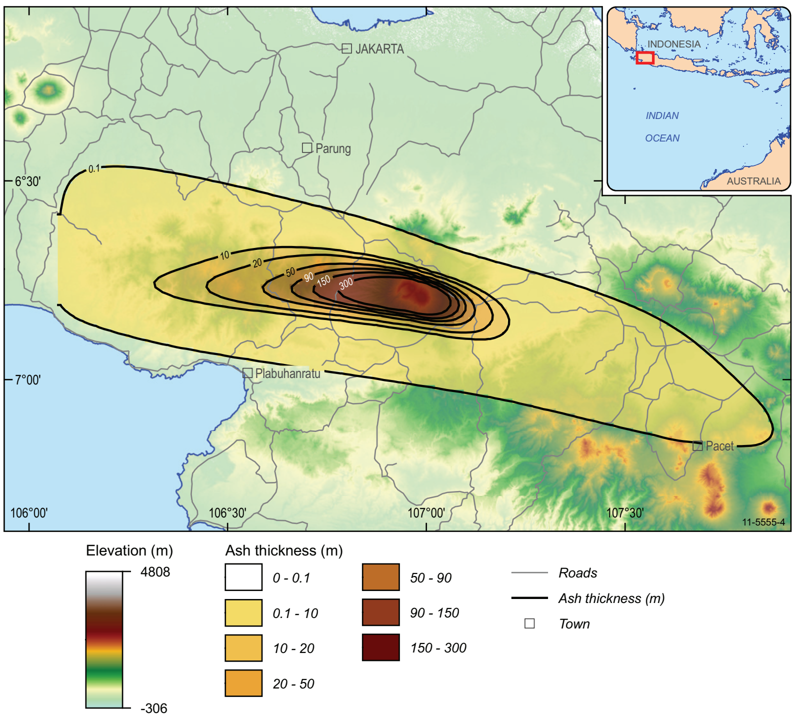


Figure 3 – A. Example python-FALL3D volcanic ash load (kg/m^2) map based on a single wind profile; **B.** Example python –FALL3D volcanic ash thickness map (m) based on a single wind profile. (Note: contours are truncated by limits of the modelled domain).

10. Open **create_hazard_map.py** using a text editor by typing:

gedit create_hazard_maps.py

11. Edit the input variables (**Table 3**).

12. Save and close.

To run type:

python create_hazard_maps.py

Outputs files are generated for each ash load threshold (*PLOAD1*, *PLOAD2* etc) in ASCII, grd, shp and kml (Google Earth) format (**Figure 4**).

To view output files navigate to the **TEPHRADATA** area:

cd /<home>/<username>/<tephra>/<volcano>

8.3 FORECASTING

This procedure details how to run a volcanological scenario using forecasted wind data produced by the BoM ACCESS-T meteorological model. Python-FALL3D downloads a 24 hour forecast, converts it into a compatible format and runs the fallout model for a projected 24 hour period - a forecasting approach.

1. Open **volcano.py** using a text editor by typing:

gedit volcano_forecast.py

7. Edit the input variables (**Table 2**; **Appendix 3**).

8. Rename the script when saving and close.

(e.g **merapi.py**)

9. To run type:

python <volcano>.py

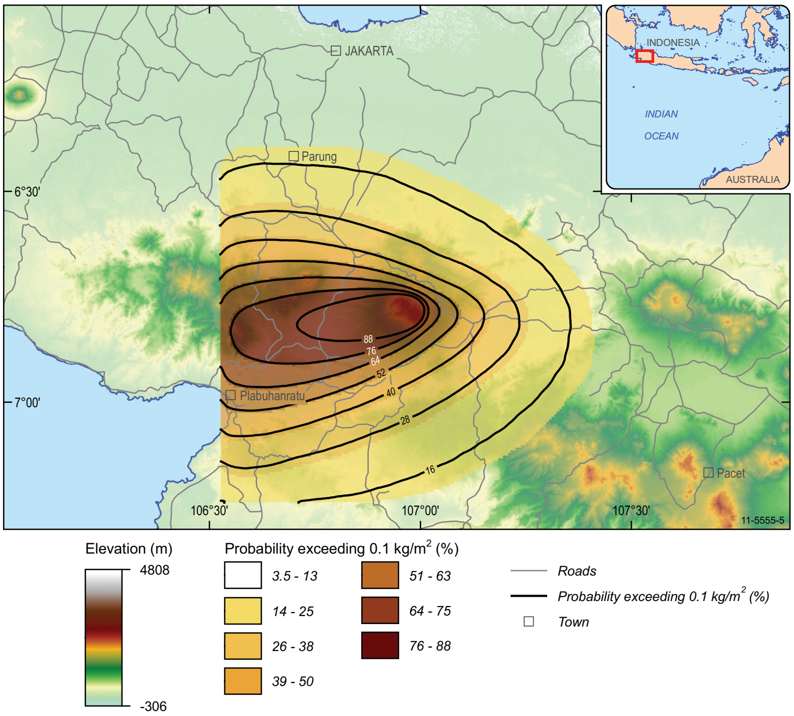
(eg. python **merapi.py**)

Outputs files are generated for each simulated hour in ASCII, grd, shp and kml (Google Earth) format (**Figure 5**).

To view output files navigate to the **TEPHRADATA** area:

cd /<home>/<username>/<tephra>/<volcano>

A.



B.

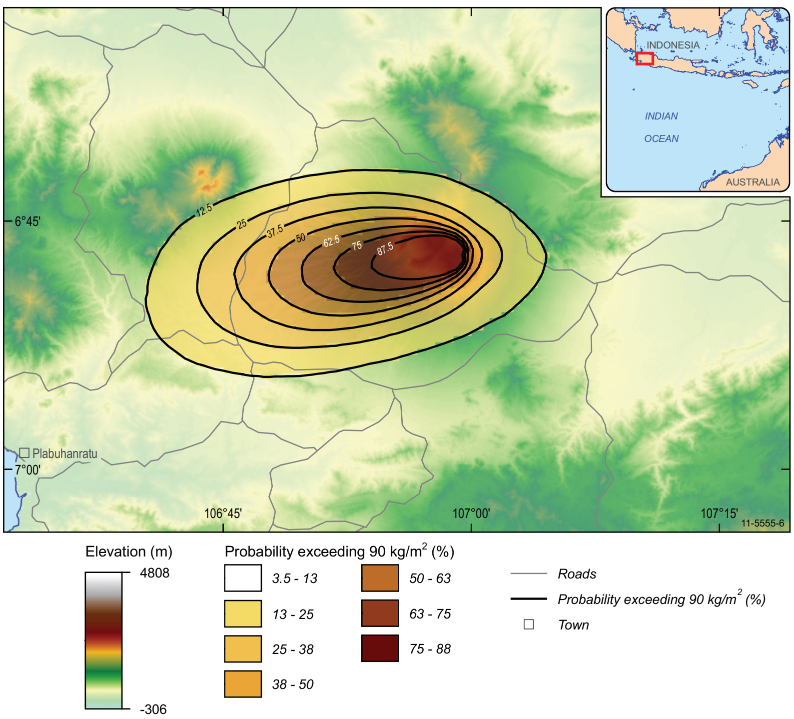
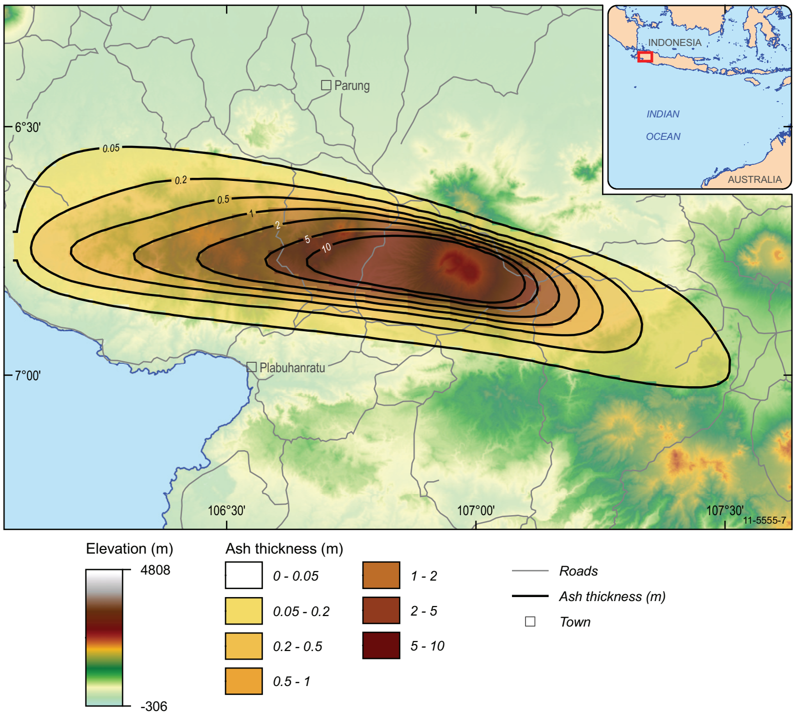


Figure 4 – Example python-FALL3D probabilistic (multiple wind) volcanic ash hazard maps for various load thresholds; **A**. 0.1 kg/m^2 (significant damage to crops; NOTE: contours are truncated by the limits of the modelled domain); **B.** 90 kg/m^2 (cosmetic damage to building exteriors).

A.



B.

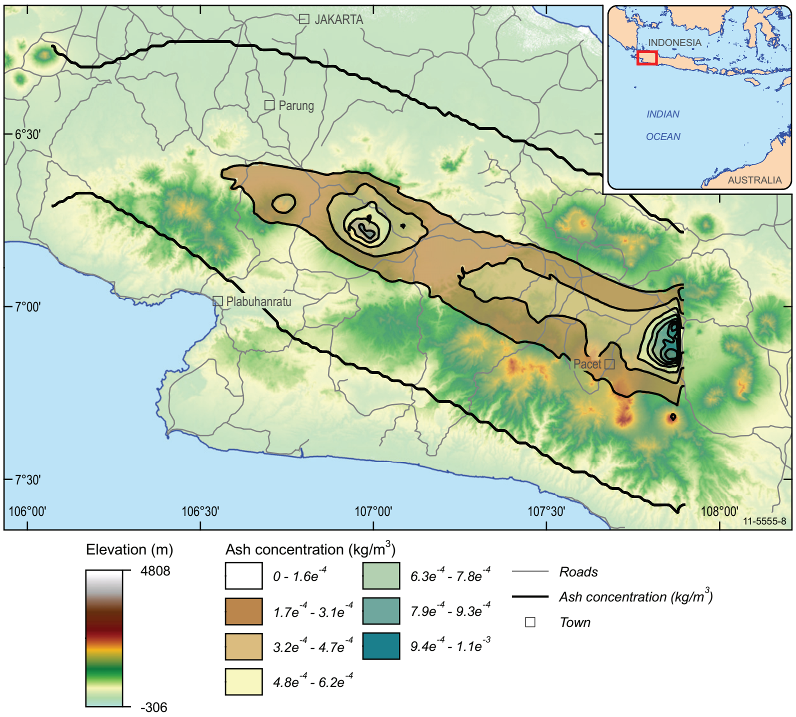


Figure 5 – A. Example python-FALL3D volcanic ash thickness map (m) based on a forecast wind profile; **B.** Example python –FALL3D volcanic ash concentration in the atmosphere map (kg/m^3) based on a forecast wind profile.

Acknowledgements

The author gratefully acknowledges technical contributions in the development of this manual made by but not limited to, O. Nielsen (AIFDR) N. Kartadinata (BG), A. Heriwaseso (BG), P.J. Delos-Reyes (PHIVOLCS), H. Mirabueno (PHIVOLCS), K. Van Putten (AIFDR). A. Folch (BSC), A. Costa (INGV) and J. Goodwin (GA). The author also acknowledges T. Dhu (AIFDR), J. Sexton (GA) and A. Simpson (GA) for feedback provided during the development of this resource. Finally, the author would like to thank colleagues in Geoscience Australia for feedback provided on a draft of this manual.

References

Barberi, F., G. Macedonio, M.T. Pareschi, and R. Santacroce, 1990. Mapping the tephra fallout risk: an example from Vesuvius, Italy, Nature, 344, 142-144.

Blong R. 2003. Building damage in Rabaul, Papua New Guinea, 1994. Bulletin of Volcanology 65, 43-54.

Blong R. and McKee C. 1995. The Rabaul eruption 1994: destruction of a town. Natural Hazards Research Centre, Macquarie University, Sydney.

Bonadonna, C. and Houghton, B.F., 2005. Total grain-size distribution and volume of tephra-fall deposits. Bulletin of Volcanology, 67: 441-456.

Carey, S.N. and Sigurdsson, H., 1982. Influence of particle aggregation on deposition of distal tephra from the May 18, 1980, eruption of Mount St-Helens volcano. Journal of Geophysical Research, 87(B8): 7061-7072.

Carey, S.N. and Sparks, R.S.J., 1986. Quantitative models of the fallout and dispersal of tephra from volcanic eruption columns. Bulletin of Volcanology, 48: 109-125.

Connor, C.B., B.E. Hill, B. Winfrey, N.M. Franklin, and P.C. LaFemina, 2001, Estimation of volcanic hazards from tephra fallout, Natural Hazards Review, 2: 33-42.

Costa, A., G. Macedonio and A. Folch, 2006. A three-dimensional Eulerian model for transport and deposition of volcanic ashes, Earth and Planetary Science Letters, 241 (3-4), 634-647.

Folch, A. and Costa, A. 2010. FALL3D-6.2 User Guide, http://www.bsc.es/projects/earthscience/fall3d/ 15 pp.

Goodwin, J. and Bear-Crozier, A. N (in prep). Volcanic ash hazard modelling using python-FALL3D: The 1994 eruption of Tavurvur, East New Britain Province, Papua New Guinea. Geoscience Australia Record.

Heffter, J.L., and B.J.B. Stunder, 1993. Volcanic Ash Forecast Transport and Dispersion (Vaftad) Model, Weather and Forecasting, 8 (4), 533-541.

Hurst, A.W., and R. Turner, 1999. Performance of the program ASHFALL for forecasting ashfall during the 1995 and 1996 eruptions of Ruapehu volcano, New Zealand Journal of Geology and Geophysics, 42 (4), 615-622.

Legros, F., 2000. Minimum volume of a tephra fallout deposit estimated from a single isopach. Journal of Volcanology and Geothermal Research, 96: 25-32.

Macedonio, G., M.T. Pareschi, and R. Santacroce, 1998. A numerical simulation of the Plinian fall phase of 79 AD eruption of Vesuvius, Journal of Geophysical Research-Solid Earth and Planets, 93 (B12), 14817-14827.

Newhall, C. G and Self, S. 1982. "The volcano explosivity index (VEI): An estimate of explosive magnitude for historical volcanism. Journal of Geophysical Research, 87 (C2), 1231-1238.

Pyle, D.M., 1989. The thickness, volume and grainsize of tephra fall deposits. Bulletin of Volcanology, 51(1): 1-15.

Searcy, C., K. Dean, and W. Stringer, 1998. PUFF: A high-resolution volcanic ash tracking model, Journal of Volcanology and Geothermal Research, 80 (1-2), 1-16.

Sulpizio, R., 2005. Three empirical methods for the calculation of distal volume of tephra-fall deposits. Journal of Volcanology and Geothermal Research, 145(3-4): 315-336.

Appendix 1 – Template for preparing digital elevation data

1. DEM File Format

```
ncols         59
nrows         64
xllcorner     412432.33601038
yllcorner     9106708.627275
cellsize      1000
NODATA_value  -9999
358 347 335 325 358 376 404 442 469 488 548 575 578 664 690 779 814 850 964 951 975 1091 1164 1234
1366   1426 1562 1586 1571 1420 1370 1296 1095 986 924 836 789 751 659 615 545 517 484 432 405 393
351 352 287
```

2. Projection File Format

```
PROJCS["WGS_1984_UTM_Zone_48S",GEOGCS["GCS_WGS_1984",DATUM["D_WGS_1984",SPHEROID["WGS_1984",6378137.0
,298.257223563]],PRIMEM["Greenwich",0.0],UNIT["Degree",0.0174532925199433]],PROJECTION["Transverse_Me
rcator"],PARAMETER["False_Easting",500000.0],PARAMETER["False_Northing",10000000.0],PARAMETER["Centra
l_Meridian",105.0],PARAMETER["Scale_Factor",0.9996],PARAMETER["Latitude_Of_Origin",0.0],UNIT["Meter",
1.
```

Appendix 2 – Preparing meteorological data

1. Navigate to the website:

http://www.esrl.noaa.gov/psd/data/reanalysis

2. Select from the list of dot-points:

"The 6-hourly and daily data currently available on-line."

3. Select from the list of blue dot points:

"Pressure Level"

The current webpage is for extracting NCEP Reanalysis 1 data at multiple **pressure levels** for a the domain.

There are 4 variables that need to be downloaded: **Air temperature**, **Geopotential Height**, **U-wind** and **V-wind**.

There are three options for each variable: **4 times daily**, **daily** and **monthly** mean.

Python-FALL3D uses the **four times daily** data (not daily or monthly).

4. Click on the coloured map for "Air Temperature (4 times daily)" to open a new webpage.
5. Click on the coloured map "Make a plot or subset" to select the region for download (i.e. Indonesia, Philippines, PNG etc) to open a new webpage.
6. Under "*Axis Dimensions*": enter the coordinates for the region that you would like to download NCEP data for: (e.g. Indonesia: lat begin: 20N, lat end: 10S, lon begin: 95E, lon end: 160E)
7. Under *"Other dimension values(s)"*: select 1000.00 millibar from the pressure level list. Hold down the shift button on your keyboard and **select all** the other pressure levels (right down to 10 millibar - this means you would like air temperature data for **every** pressure level)
8. Select the date you wish to download in UTC time.
9. Under *"Output options"*: select "Create a subset without making a plot"
10. Under *"Plot output options*": deselect "Color plot"
11. Select "Create Plot or Subset of Data" to open a new webpage.
12. Select "FTP copy of the file"
13. Save the file when prompted
14. Return to the webpage with the first colour map (Step 4).
15. Repeat steps 4-14 for the three remaining variables "Geopotential Height", "u-wind" and "v-wind".
16. There should be 4 files with the extension **.nc** at the conclusion of the download process
17. Rename the Air Temperature file "**TMP.nc**"
18. Rename the Geopotential Height file "**HGT.nc**"
19. Rename the u-wind file "**UGRD.nc**"
20. Rename the v-wind file "**VGRD.nc**"
21. Note the pathway to the directory where the files are stored for modelling purposes

e.g. '/<**model_area**>/<**tephra**>/<**NCEP**>'

Appendix 3 – Volcanological input worksheet

 # Short eruption comment to appear in output directory.
eruption_comment = ______________________

#Temporal parameters

eruption_start =__________________________# Hours relative to the start of wind data
eruption_duration =_______________________# Hours
post-eruptive_settling_duration =_____________# Hours (to allow for ash settling)

 # Location
x_coordinate_of_vent = ___________________# UTM zone implied by projection
y_coordinate_of_vent =____________________# UTM zone implied by projection

 # Vertical discretisation for model domain
z_min = ________________________________
z_max = _______________________________
z_increment = ___________________________

 # Meteorological input (Refer to Table 2)
wind_profile = ___________________________# Path to wind data or online forecasts

 # DEM model (Refer to Table 2)
topography_grid = _________________________# Path to topography file

 # Granulometry
grainsize_distribution = ___________________# Possibilities are GAUSSIAN/BIGAUSSIAN(modal/bimodal)
number_of_grainsize_classes = _____________
mean_grainsize = _________________________# phi
sorting = ________________________________
minimum_grainsize = ______________________# phi
maximum_grainsize = ______________________# phi
density_minimum = ________________________# kg/m3
density_maximum = ________________________# kg/m3
sphericity_minimum = ______________________
sphericity_maximum =______________________

 # Source
vent_height = _____________________________# meters
source_type = _____________________________# Possibilities are **'plume', 'suzuki', 'point'**
mass_eruption_rate = _______________________# kg/s (if unknown **'estimate')**
height_above_vent = ________________________# m
A = _____________________________________# (suzuki only)
L = _____________________________________# (suzuki only)
height_or_MFR = ___________________________# plume only
MFR_minimum = ____________________________# kg/s (plume only)
MFR_maximum =____________________________# kg/s (plume only)
exit_velocity = ____________________________# m/s (plume only)
exit_temperature = _________________________# K (plume only)
exit_volatile_fraction = ______________________# % (plume only)

 # Fall3D
terminal_velocity_model = __________________# Possibilities are ARASTOOPOR/GANSER/WILSON/DELLINO
vertical_turbulence_model = ________________# Possibilities are CONSTANT/SIMILARITY
horizontal_turbulence_model = ______________# Possibilities are CONSTANT/RAMS
vertical_diffusion_coefficient = _______________# m2/s
horizontal_diffusion_coefficient = _____________# m2/s
value_of_CS = ____________________________# RAMS only

 # Contouring
thickness_contours = ______________________# if unknown ‘True’
load_contours = ___________________________# kg/m2
thickness_units = _________________________# mm, cm, m

Appendix 4 – Range table

The range table below details the acceptable range of eruption column heights, eruption column increments, mass eruption rates and eruption durations that should be adhered to when considering a new scenario. The table is based on the volcano explosivity index (VEI; Newhall and Self, 1982).

Ranges	VEI 2	VEI 3	VEI 4	VEI 5	VEI 6	VEI 7	VEI 8
Eruption column height (m)	2000-5000	3000-15000	10000-25000	25000-30000	30000-50000	30000-50000	50000+
Eruption column height increments (m)[#]	1000	1000-10000	10000	10000	10000	10000	10000
Mass eruption rates (kg/s)	$1x10^4$-$1x10^6$	$1x10^4$-$1x10^6$	$1x10^5$-$1x10^8$	$1x10^5$-$1x10^8$	$1x10^9$-$1x10^{12}$	$1x10^9$-$1x10^{12}$	$1x10^9$-$1x10^{15}$
Eruption duration (hours)	1-6	1-6	1-6	6-12	>12	>12	>12

[#] *Increments must always be the same magnitude of order as the eruption column height (i.e. 3,000m (1,000 increments; 40,000m (10,000 increments)*

Appendix 5 – Running in parallel (multiple nodes)

The command below shows an example host file for a cluster with 20 dual-cpu quad core nodes, i.e. 8 processes can run on each node.

mpirun -x FALL3DHOME -x PYTHONPATH -hostfile /etc/mpihosts -host node<#>,node<#> python <volcano>.py

A host file for the system must be specified for this command to work. The file must contain the names of each computer node in the system along with information about how many processes can run independently on each node.

For more details on hostfiles see
e.g. **http://linux.die.net/man/1/mpirun or http://www.open-mpi.org/faq/?category=running**

Appendix 6 - Glossary of volcanological and meteorological terms

Point

Mass of an eruption column is released at a single source point (Folch and Costa, 2010)

Suzuki

Mass of an eruption column released according to an empirically derived formula (Folch and Costa, 2010)

Plume

Mass of an eruption column released according to the buoyant plume theory. (Folch and Costa, 2010)

Rams/ Constant (horizontal)

Equations for solving the horizontal diffusion co-efficient of settling particles. (Folch and Costa, 2010)

Similarity/ Constant (vertical)

Equations for solving the vertical diffusion co-efficient of settling particles.
(Folch and Costa, 2010)

ARASTOOPUR

Mathematical formula for estimating the settling velocity of particles.
(Folch and Costa, 2010)

GANSER

Mathematical formula for estimating the settling velocity of particles.
(Folch and Costa, 2010)

WILSON

Mathematical formula for estimating the settling velocity of particles.
(Folch and Costa, 2010)

DELLINO

Mathematical formula for estimating the settling velocity of particles.
(Folch and Costa, 2010)